New results on Monte Carlo bit error simulation based on the a posteriori log-likelihood ratio

Land, Ingmar; Hoeher, Peter A.

Published in:
Proc. Int. Symp. on Turbo Codes & Rel. Topics, Brest, France, Sept.

Publication date:
2003

Document Version
Publisher's PDF, also known as Version of record

New Results on Monte Carlo Bit Error Simulation Based on the A Posteriori Log-Likelihood Ratio

Ingmar Land and Peter A. Hoeher

Information and Coding Theory Lab,
Faculty of Engineering, University of Kiel, Germany
www.tf.uni-kiel.de/ict, {il,ph}@tf.uni-kiel.de

Abstract: *There are two methods for estimating the bit error probability of a transmission system via Monte Carlo simulation, when the decoder outputs a-posteriori log-likelihood ratios (LLR). The first method, which is the conventional one, is based on the sign of the LLR, whereas the second method is based on the magnitude of the LLR. In this paper, the two methods are compared by means of their estimation variances. Furthermore, the optimal linear combination of the two methods is considered. The superiority of second method over the first one will be proven.*

Keywords: APP decoding, log-likelihood ratio, Monte Carlo simulation, bit error rate.

1. Introduction

The bit error rate is one of the most important quality criteria for digital transmission systems. As many systems are too complex for an analytical derivation, Monte Carlo simulation is often applied for bit error rate estimation. If the transmission system employs a LogAPP decoder [1], which outputs the a-posteriori log-likelihood ratio (LLR) of each bit, there are two different methods available. (This holds similarly for an APP decoder [2].)

The first method, referred to as Method H, relies solely on the signs of the a-posteriori LLRs. It is conventionally applied in bit error simulation. The second method, referred to as Method S, relies solely on the magnitudes of the a-posteriori LLRs and does not require knowledge of the transmitted bits. This method was first published by Loeliger [3], and was then independently re-invented and further analyzed in [4].

Based on a formal definition of the two methods, we will address two aspects in this paper: (a) The two methods will be compared with respect to their estimation variance. We will show that the estimation variance of Method S is at most half the estimation variance of Method H. This proves that Method S is superior to Method H. (b) The optimal linear combination of the two methods will be derived. We will show that this combination is identical to Method S. Thus, Method S cannot be further improved by linearly combining it with Method H.

2. Transmission System

The transmission system and the simulation setup under consideration is depicted in Figure 1. Throughout this paper random variables will be denoted by uppercase letters and their realizations by the corresponding lowercase letters.

The sequence of independent and identically distributed binary information symbols $u \in \{+1, -1\}$ is encoded by the channel encoder (ENC) onto the sequence of code symbols x. These are transmitted over the channel (CH), and the vector of channel outputs $\boldsymbol{y}$ is fed to the channel decoder (LogAPP DEC). Note that this system is very general: (i) The information symbols are not required to be uniformly distributed. (ii) The channel code does not need to be binary, not even linear. (iii) The channel may be frequency selective, time varying, or even nonlinear.

Given $\boldsymbol{y}$, the channel decoder computes the a-posteriori LLR $l \in \mathbb{R}$ for each information bit U,

$$l := L(U|\boldsymbol{Y} = \boldsymbol{y}) := \ln \frac{P_{U|\boldsymbol{Y}}(+1|\boldsymbol{y})}{P_{U|\boldsymbol{Y}}(-1|\boldsymbol{y})},$$

[5], [6]. (Throughout this paper, the indices will be omitted for convenience, whenever this can be done without ambiguity.) For "simple" systems, like convolutional encoded transmission over an AWGN channel or transmission with DPSK over an ISI channel, these LLRs can be efficiently computed by means of the LogAPP algorithm (LogMAP algorithm [1], BCJR algorithm [2], forward-backward algorithm). The a-posteriori LLR l will be regarded as a realization of the random variable L. This soft output is then separated into its sign $\hat{u} := \mathrm{sign}(l)$ and its magnitude $\lambda := \mathrm{abs}(l)$. The sign represents the hard decision and thus the estimate for the transmitted information bit u, and the magnitude represents the reliability.

3. Monte Carlo BER Estimation

For determining the bit error rate (BER) of the transmission system, $P_b := \Pr(U \neq \hat{U})$, two kinds of BER samples are considered in this paper. The *hard BER sample* is defined as

$$z_H := \Pr(U \neq \hat{U}|U = u, \hat{U} = \hat{u}) \tag{1}$$

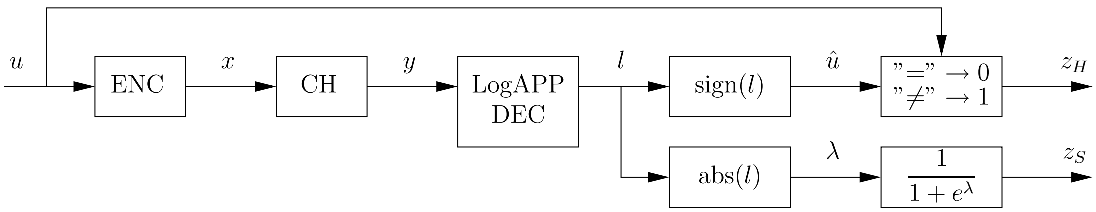

Figure 1: Transmission system and simulation setup.

and indicates *whether* a bit error occurred or not. It is easily seen that

$$z_H = \begin{cases} 0 & \text{if } u = \hat{u}, \\ 1 & \text{if } u \neq \hat{u}. \end{cases} \tag{2}$$

This BER sample relies solely on the sign of the soft output; additionally, knowledge of the transmitted bit is required. The *soft BER sample* is defined as

$$z_S := \Pr(U \neq \hat{U} | \Lambda = \lambda) \tag{3}$$

and indicates the *probability* that a bit error occurred for a given value of λ. It can be computed as

$$z_S = \frac{1}{1 + e^{\lambda}}, \tag{4}$$

[4], with $z_S \in [0, \frac{1}{2}]$. This BER sample relies solely on the magnitude of the soft output; knowledge of the transmitted information bit is not necessary.

Given J LLRs l_j and thus the corresponding BER samples $z_{H,j}$ and $z_{S,j}$, $j = 1, 2, \ldots, J$, the two methods for estimating the BER are as follows:

Method H: The BER estimate based on J hard BER samples $z_{H,j}$ is computed as

$$z_H^{(J)} := \frac{1}{J} \sum_{j=1}^{J} z_{H,j},$$

and it is denoted as *hard BER estimate*.

Method S: The BER estimate based on J soft BER samples $z_{S,j}$ is computed as

$$z_S^{(J)} := \frac{1}{J} \sum_{j=1}^{J} z_{S,j},$$

and it is denoted as *soft BER estimate*.

The first method basically consists of counting the occurred bit errors and averaging over the number of transmitted bits. This is the conventional method and works for every decoder. As opposed to this, the second method relies on the fact that the outputs of the decoder are a-posteriori log-likelihood ratios. Therefore, it is only applicable for LogAPP decoders (or equivalently for APP decoders).

In the following, the hard and the soft BER sample will be regarded as random variables $Z_H \in \{0, 1\}$ and $Z_S \in [0, \frac{1}{2}]$, respectively. From their definitions, it follows that $\mu = \mathrm{E}\{Z_H\} = \mathrm{E}\{Z_S\} = P_b$. Since the hard and the soft BER estimate are sample means, it follows also that $\mathrm{E}\{Z_H^{(J)}\} = \mathrm{E}\{Z_S^{(J)}\} = P_b$. Thus, both estimates are unbiased.

4. Comparison and Combination of the Two Methods

4.1. Variances

In both methods, the BER estimate is a sample mean. Thus, an appropriate figure-of-merit is the estimation variance. Let $\sigma^2_{Z_H}$ and $\sigma^2_{Z_S}$ denote the variances of Z_H and Z_S, and let further $\sigma^2_{Z_H^{(J)}}$ and $\sigma^2_{Z_S^{(J)}}$ denote the variances of the sample means $Z_H^{(J)}$ and $Z_S^{(J)}$, respectively. The variances of the (single) samples and those of the sample means are related as

$$\sigma^2_{Z_H^{(J)}} = \frac{1}{J} \sigma^2_{Z_H}, \qquad \sigma^2_{Z_S^{(J)}} = \frac{1}{J} \sigma^2_{Z_S}.$$

Let Z_X denote (generically) either Z_H or Z_S. A suitable measure for the precision of the estimation is the relative standard deviation of the sample mean. For a given variance $\sigma^2_{Z_X}$, the relation between the precision, ε, and the number of samples, J, is given by

$$\varepsilon = \frac{\sigma_{Z_X^{(J)}}}{\mu} = \sqrt{\frac{\sigma^2_{Z_X}}{\mu^2 \cdot J}}. \tag{5}$$

This expression will be used to compare the two methods with respect to the achievable precision for a fixed number of samples or with respect to the necessary number of samples to achieve a certain precision.

4.2. Probability Distributions

The hard BER sample is distributed as

$$p_{Z_H}(z_H) = \begin{cases} 1 - \mu & \text{for } z_H = 0, \\ \mu & \text{for } z_H = 1. \end{cases} \tag{6}$$

Given a soft BER sample z_S, the conditional distribution can be written as

$$p_{Z_H|Z_S}(z_H|z_S) = \begin{cases} 1 - z_S & \text{for } z_H = 0, \\ z_S & \text{for } z_H = 1. \end{cases} \tag{7}$$

The distribution $p_{Z_S}(z_S)$ of the soft BER sample strongly depends on the system under consideration; thus, the only known property is the mean value μ.

4.3. Statistical Dependence

For measuring the statistical dependence, we will compute the mutual information between the hard and the soft BER sample. Let $h(x) = -x\,\mathrm{ld}\,x - (1-x)\,\mathrm{ld}\,(1-x)$, $x \in [0,1]$, denote the binary entropy function. Then we have

$$\begin{aligned} I(Z_H; Z_S) &= H(Z_H) - H(Z_H|Z_S) \\ &= H(Z_H) - \underset{z_S}{\mathrm{E}}\{H(Z_H|Z_S = z_S)\} \\ &= h(\mu) - \mathrm{E}\{h(Z_S)\}. \end{aligned}$$

In the last line, we applied $H(Z_H|Z_S = z_S) = h(p_{Z_H|Z_S}(1|z_S)) = h(z_S)$, where (7) was used. The binary entropy function can be bounded as $h(z_S) \geq 2z_S$, and thus we can bound the mutual information as

$$I(Z_H; Z_S) \leq h(\mu) - 2\mu.$$

Therefore, we can conclude that there is statistical dependence, but that this dependence is very small for small BER. It even tends to zero for $P_b \to 0$.

4.4. Comparison

The variance of the hard BER sample Z_H can be written as

$$\begin{aligned} \sigma^2_{Z_H} &= \mathrm{E}\{Z_H^2\} - \mu^2 = \\ &= \mathrm{E}\{Z_H\} - \mu^2 = \mu(1-\mu), \end{aligned} \tag{8}$$

where the identity $Z_H^2 = Z_H$ was applied. The variance of the soft BER sample Z_S can be written as

$$\sigma^2_{Z_S} = \mathrm{E}\{Z_S^2\} - \mu^2. \tag{9}$$

Further simplification is not possible (cf. comments on the distribution of Z_S). In the following, two lower bounds on the ratio of these variances will be derived.

From $Z_S \in [0, \frac{1}{2}]$, it follows that $Z_S^2 \leq \frac{1}{2} Z_S$, and thus

$$\mathrm{E}\{Z_S^2\} \leq \tfrac{1}{2}\,\mathrm{E}\{Z_S\}. \tag{10}$$

This inequality will be the starting point for the two bounds. Note that equality holds for $Z_S = \frac{1}{2}$ and for $Z_S = 0$, which corresponds to $P_b = \frac{1}{2}$ and $P_b = 0$, respectively. In each case, Z_S is constant, and thus its variance is zero. As this is not of interest, we will assume $0 < P_b < \frac{1}{2}$ in the following.

For deriving the first bound, we write the left hand side of (10) as $\mathrm{E}\{Z_S^2\} = \sigma^2_{Z_S} + \mu^2$ and the right hand side as

$$\mathrm{E}\{Z_S\} = \mathrm{E}\{Z_H\} = \mathrm{E}\{Z_H^2\} = \sigma^2_{Z_H} + \mu^2.$$

Using these two equalities in (10) yields

$$\begin{aligned} & & \sigma^2_{Z_S} + \mu^2 &\leq \tfrac{1}{2}(\sigma^2_{Z_H} + \mu^2) \\ &\Leftrightarrow & \sigma^2_{Z_S} &\leq \tfrac{1}{2}\sigma^2_{Z_H} - \tfrac{1}{2}\mu^2 \\ &\Rightarrow & \sigma^2_{Z_S} &\leq \tfrac{1}{2}\sigma^2_{Z_H}, \end{aligned} \tag{11}$$

where equality holds in the last line if and only if $\mu = P_b = 0$. Thus, we have the first bound b_1:

$$\frac{\sigma^2_{Z_H}}{\sigma^2_{Z_S}} \geq 2 := b_1. \tag{12}$$

For deriving the second bound, we write the left hand side of (10) as before, and we substitute $\mathrm{E}\{Z_S\} = \mu$ on the right hand side. Thus, we have

$$\begin{aligned} & & \sigma^2_{Z_S} + \mu^2 &\leq \frac{\mu}{2} \\ &\Leftrightarrow & \sigma^2_{Z_S} &\leq \frac{\mu(1-2\mu)}{2}. \end{aligned}$$

Using this inequality and (8), the ratio of the variances can be written as

$$\frac{\sigma^2_{Z_H}}{\sigma^2_{Z_S}} \geq \frac{2\mu(1-\mu)}{\mu(1-2\mu)} = \frac{2-2\mu}{1-2\mu} = \frac{2-2P_b}{1-2P_b},$$

and we have the second bound $b_2(P_b)$,

$$\frac{\sigma^2_{Z_H}}{\sigma^2_{Z_S}} \geq \frac{2-2P_b}{1-2P_b} := b_2(P_b), \tag{13}$$

which is a function of P_b. This second bound $b_2(P_b)$ is tighter than the first bound b_1 for large P_b, and it is equal to the first bound for small P_b.

The bounds shall be illustrated in the following example: Uniformly distributed information bits are encoded by a convolutional encoder of rate 1/2 and memory length ν, and the code bits are transmitted over a binary-input AWGN channel. (E_b denotes the signal energy per information bit, N_0 denotes the single-sided noise power density.) The resulting ratios of variances and the two bounds are plotted versus the signal-to-noise ratio (SNR) in Figure 2. For low SNR, both the actual ratio $\sigma^2_{Z_H}/\sigma^2_{Z_S}$ and the bound $b_2(P_b)$ are decreasing. For higher SNR, $b_2(P_b)$ tends to $b_1 = 2$, and $\sigma^2_{Z_H}/\sigma^2_{Z_S}$ tends to about 4.

The bounds prove that the hard BER sample has always (except for $P_b = 0$) a larger variance than the soft BER sample. Moreover, it follows immediately from (12) and (13) that

$$\frac{\sigma^2_{Z_H^{(J)}}}{\sigma^2_{Z_S^{(J)}}} \geq b_2(P_b) \geq b_1 = 2,$$

where again equality holds if and only if $P_b = 0$. Taking (5) into account, the advantage of Method S over Method H becomes obvious and can be formulated in the following two (equivalent) ways: (a) For achieving a required precision of the BER estimate, Method S needs only half the number of samples or even less. (b) Given a fixed number of samples, the precision achieved by Method S is higher by a factor of a least $\sqrt{2}$.

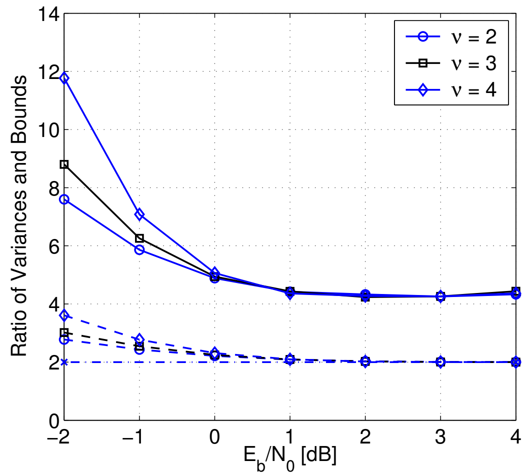

Figure 2: Ratios of variances $\sigma^2_{Z_H}/\sigma^2_{Z_S}$ (solid lines) and the lower bounds b_1 (dash-dotted line) and $b_2(P_b)$ (dashed lines) versus E_b/N_0. (Uniformly distributed information bits, convolutional encoder of rate 1/2 and memory length ν, binary-input AWGN channel.)

4.5. Optimal Linear Combination

Since Method H relies solely on the sign of the soft output and Method S relies solely on the magnitude of the soft output, the question arises if these two methods can be combined to get a BER estimate even better than the soft BER estimate.

In the following, we will consider the optimal linear combination of the hard and the soft BER sample, having minimum variance. Let define the combined BER sample

$$z := az_H + (1-a)z_S \tag{14}$$

with weighting factor $a \in [0,1]$. From its definition, it follows immediately that Z is unbiased, i.e., $\mathrm{E}\{Z\} = \mu = P_b$.

Let now determine a such that the variance of Z is minimum. This variance can be written as

$$\begin{aligned}\sigma^2_Z &= \mathrm{E}\{(Z-\mu)^2\} \\ &= \mathrm{E}\{(aZ_H + (1-a)Z_S - \mu)^2\} \\ &= \mathrm{E}\{(a(Z_H-\mu) + (1-a)(Z_S-\mu))^2\} \\ &= a^2\sigma^2_{Z_H} + (1-a)^2\sigma^2_{Z_S} + 2a(1-a)\sigma^2_{Z_H Z_S}.\end{aligned}$$

The conditional expectation of Z_H computes as

$$\mathrm{E}\{Z_H | Z_S = z_S\} = \sum_{z_H \in \{0,1\}} p_{Z_H|Z_S}(z_H|z_S) z_h = z_s,$$

where (7) was applied. Thus, the covariance can be expressed as

$$\begin{aligned}\sigma^2_{Z_H Z_S} &= \mathrm{E}\{(Z_H-\mu)(Z_S-\mu)\} \\ &= \mathrm{E}\{Z_H Z_S\} - \mu^2 \\ &= \underset{Z_S}{\mathrm{E}}\{z_S \underset{Z_H|Z_S=z_S}{\mathrm{E}}\{z_H\}\} - \mu^2 \\ &= \underset{Z_S}{\mathrm{E}}\{z_S^2\} - \mu^2 \quad = \quad \sigma^2_{Z_S}.\end{aligned}$$

With this result, the variance of Z can be written as

$$\begin{aligned}\sigma^2_Z &= a^2\sigma^2_{Z_H} + (1-a)^2\sigma^2_{Z_S} + 2a(1-a)\sigma^2_{Z_S} \\ &= a^2\sigma^2_{Z_H} + (1-a^2)\sigma^2_{Z_S}.\end{aligned} \tag{15}$$

Note that this expression contains only the variances of the hard and the soft BER sample and the weighting factor a.

The extremum is found by evaluating

$$\frac{d}{da}\sigma^2_Z = 2a\sigma^2_{Z_H} - 2a\sigma^2_{Z_S} \stackrel{!}{=} 0.$$

The single solution is $a = 0$, because $\sigma^2_{Z_H} > \sigma^2_{Z_S}$ for $P_b > 0$. (The case $P_b = 0$ is not of interest.) Due to the same reason, the second derivative is strictly positive:

$$\frac{d^2}{da^2}\sigma^2_Z = 2\sigma^2_{Z_H} - 2\sigma^2_{Z_S} > 0.$$

Thus, we get the minimum variance of Z for $a = 0$.

From this result, it follows that the optimum linear combination of the hard and the soft BER sample w.r.t. minimum estimation variance consists only of the soft BER sample.

REFERENCES

[1] P. Robertson, P. Hoeher, and E. Villebrun, "Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding," *Europ. Trans. Telecommun.*, vol. 8, no. 2, pp. 119–125, Mar./Apr. 1997.

[2] L. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, pp. 284–287, Mar. 1974.

[3] H. A. Loeliger, "A posteriori probabilities and performance evaluation of trellis codes," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Trondheim, Norway, June 1994, p. 335.

[4] P. Hoeher, I. Land, and U. Sorger, "Log-likelihood values and Monte Carlo simulation - some fundamental results," in *Proc. Int. Symp. on Turbo Codes & Rel. Topics*, Brest, France, Sept. 2000, pp. 43–46.

[5] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Trans. Inform. Theory*, vol. 42, no. 2, pp. 429–445, Mar. 1996.

[6] I. Land, P. Hoeher, and U. Sorger, "On the interpretation of the APP algorithm as an LLR filter," in *Proc. IEEE Int. Symp. Inform. Theory (ISIT)*, Sorrento, Italy, June 2000, p. 415.